Lab: Approximations for hard problems

Submit the solutions to the following problems to PolyLearn as a pdf file.

1. Solve the following the minimum cost assignment of jobs to people using a greedy algorithm that at each step chooses the lowest cost person-job pair that is feasible. The first person-job pair would be P_3-Job$_3$, then find the lowest cost assignment that does not conflict with any previous assignment.
   A. Show the assignments in the order in which they are found by the greedy algorithm.
   B. Write the pseudo code for the algorithm?
   C. What is its complexity as a function of n?

	Job$_1$	**Job $_2$**	**Job $_3$**	**Job $_4$**
P_1	9	2	7	8
P_2	8	4	3	9
P_3	4	8	1	8
P_4	7	5	9	6

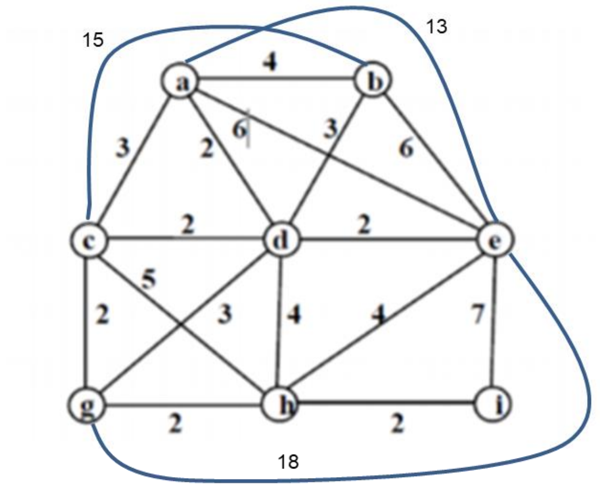
2. Use Branch and Bound to find the minimum cost assignment of jobs to people for the problem in #1. That is, draw the state space tree showing the order in which the nodes are created in the state space tree, see page 435 in the text. As in the text, for each interior node in the tree show the current assignment and the lower bound and for each leaf the assignment and the total cost.

3. Apply a greedy algorithm based on ***greatest w_i / v_i first*** to solve the Knapsack problem using the bounding function discussed in class. Capacity of the Knapsack = 70

	Item$_1$	**Item$_2$**	**Item$_3$**	**Item$_4$**
w_i	10	30	20	70
v_i	11	30	19	63
w_i / v_i	1.1	1.0	0.95	0.9

4. Apply Branch and Bound to solve the same Knapsack problem using the bounding function discussed in class. Capacity of the Knapsack = 70

5. Find an approximate solution to the TSP problem for the following graph using “Twice Around the Tree” starting from vertex a. Why might this not be a 2 approximation?


15
13
4
a
b
3
2
6
3
6
c
d
e
2
2
5
2
3
4
4
7
g
h
i
2
2
18